UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/781,512	02/17/2004	Chee Siong Lee	42P18829	2960

8791 7590 01/25/2007

BLAKELY SOKOLOFF TAYLOR & ZAFMAN
12400 WILSHIRE BOULEVARD
SEVENTH FLOOR
LOS ANGELES, CA 90025-1030

EXAMINER	
UNELUS, ERNEST	
ART UNIT	**PAPER NUMBER**
2181	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/25/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/781,512	LEE ET AL.	
	Examiner	Art Unit	
	Ernest Unelus	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 February 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18 and 20-24* is/are pending in the application.

4a) Of the above claim(s) *19* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18 and 20-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *17 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. The instant application having Application No. 10/781,512 has a total of 24 preliminary amended claims pending in the application; there are 4 independent claims and 20 dependent claims, all of which are ready for examination by the examiner.

I. INFORMATION CONCERNING OATH/DECLARATION

Oath/Declaration

2. The applicant's oath/declaration has been reviewed by the examiner and is found to conform to the requirements prescribed in **37 C.F.R. 1.63.**

II. INFORMATION CONCERNING DRAWINGS

Drawings

3. The applicant's drawings submitted are acceptable for examination purposes.

III. REJECTIONS BASED ON PRIOR ART

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:
(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The indicated allowability of claims 1-10 are withdrawn in view of the newly discovered reference(s) to Bernasconi et al. (US pat. 6,158,018). Rejections based on the cited reference(s) follow.

4 **Claims 1-18 and 20-24** are rejected under 35 U.S.C. 103(a) as being unpatentable over Bernasconi et al. (US pat. 6,158,018) in view of Hagan et al. (US pat. 5,966,547).

5. As per **claims 1 and 10**, Bernasconi discloses a system **(system 10 of fig. 1)** comprising: an input/output (I/O) controller **(microcontroller 14 of fig. 1, as discloses in col. 4, lines 44 and 45)** to receive request cycles **(the DSP program address, as it read,,-see col. 9, line 54)** from a requesting device **(DSP 16 of fig. 1)**; and a patch module **(patching circuitry 22 of fig. 1)**, the patch module to capture an incoming cycle **(see fig. 1)** received by the I/O controller and to determine if the captured incoming cycle matches one or more of preprogrammed trigger conditions **(the comparator 42 of the patching circuitry 22, as discloses in col. 7, lines 16-19) (col. 9, lines 51-57, teaches that the trigger-matching logic is the logic that doest the matching between the current DSP program address and a break address correspond to a flawed DSP program stored in the ROM 18, such as at section 18b)**; wherein the patch module can work around a captured non-posted request cycle **(a faulty DSP program)** by controlling header information **(the fault address of the DSP program) (col. 8, lines 3-11 discloses "When the patch control module 60 is enabled over line 64, and the comparator 42 detects a match, the signal indicating a match over line 62 causes the patch control module 60 to initiate a patch sequence. Initiation of a patch sequence will render appropriate signals over the patch control outputs labeled patch.cycle 1 on line 74 and patch.cycle 2 on line 76. Details of patch sequence initiation will be discussed later with**

respect to device operation". This 'Operation', as sated in col. 9, line 57 to col. 10, line 5, discloses, "At this stage, the patching circuitry 22 sends to the DSP 16 a branch op code followed by a branch address corresponding to the beginning of a block of corrected DSP program software in the RAM 20 such as in section 20b thereof. Therefore, in the third general phase of device operation, the DSP 16 fetches corrected DSP program software instructions from the RAM 20 such as section 20b thereof. Moreover, corrected DSP program software can be set up to cause the DSP 16 to establish new break and branch addresses, thereby setting up additional patches to be executed during the operation of the device's DSP 16. Also, a section of corrected DSP program software typically includes, at the end thereof, jumping instructions causing the DSP 16 to recommence fetching DSP program software from the ROM 18 in a section such as 18c located downstream of the previous flawed portion 18b in the ROM 18"). In regards to "a completion packet received from a designated end-device being discard" **(directionally, fig. 1 shows the DSP program address coming from the DSP 16,,,the claim language doesn't preclude the DSP 16 from being 'a requesting device' and 'an end-device'. The claim language doesn't specifically express that the requesting device is different from an end-device). Once an incoming cycle is received from a requesting device, it is considered complete until an error or a fault is found in it; as it goes through the patch circuitry 22, it will be discard because of the error or the fault. Therefore, this limitation is inherently met.**

Bernasconi fails to specifically teach specifying whether the I/O controller including a patch module **(patching circuitry22)** coupled to a completion queue is to be loaded with information from non-posted cycle.

Hagan discloses "a RAM or ROM wherein a completion queue is to be loaded with information from non-posted cycle" **(see abst. and col. 2, lines 50-65). See abst. and col. 4, lines 47-59, which also discloses an interrupt state to be cleared when the entry is posted, the interrupt state being associated with each entry in the queue.**

Bernasconi's invention and Hagan's invention's are analogous art because they are from the same field of endeavor storing data into queue from a processor. In view of such teaching, at the time of the invention it would have been obvious to a person of ordinary skill in the art to modify the patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip as taught by Bernasconi which includes a DSP and a patching circuitry that's consist of a trigger-matching logic, further include a control logic coupled to the trigger-matching logic is modified to include a completion queue to be loaded with information from non-posted cycle as taught by Hagan.

The motivation for doing so would have been because Hagan teaches that **["The significant instructions generally translate into increased processing time, slowing the response of the processor determining whether the current queue entry is empty. Therefore, it would be advantageous to have an improvement with an apparatus for reducing the processing overhead for multiple processor or embedded processor architectures in posting events or tasks to a queue"(col. 1, lines 49-56)]**.

Therefore, it would have been obvious to combine Hagan and Bernasconi for the benefit of creating a patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip to obtain the invention as specified in

claims 1 and 10.

6. As per **claims 2, 5 and 8**, Bernasconi discloses "the system of claim 1,"**[see rejection to claim 1 above]** "wherein if a captured non-posted cycle causes a patch trigger **(col. 9, lines 51-57, teaches that the trigger-matching logic is the logic that doest the matching between the current DSP program address and a break address correspond to a flawed DSP program stored in the ROM 18, such as at section 18b.) to capture an incoming cycle, which is a non-posted cycle (col. 9, line 54 teaches the current DSP program address corresponds to the applicant incoming cycle).**

In regards to "the patch module can load the completion queue with modified and unmodified header information obtained from the captured non-posted cycle", Bernasconi fails to specifically teach whether the I/O controller including a patch module **(patching circuitry22)** coupled to a the completion queue with modified and unmodified header information obtained from the captured non-posted cycle.

Hagan discloses "a RAM or ROM wherein a completion queue is to be loaded with modified and unmodified header information obtained from the captured non-posted cycle" **(see abst. and col. 2, lines 50-65). See abst. and col. 4, lines 47-59, which also discloses an interrupt state to be cleared when the entry is posted, the interrupt state being associated with each entry in the queue.**

Bernasconi's invention and Hagan's invention's are analogous art because they are from the same field of endeavor storing data into queue from a processor. In view of such teaching, at the time of the invention it would have been obvious to a person of ordinary skill in the art to modify the patch mechanism used to detect and workaround

defects and conditions existing in an integrated circuit chip as taught by Bernasconi which includes a DSP and a patching circuitry that's consist of a trigger-matching logic, further include a control logic coupled to the trigger-matching logic is modified to include a completion queue to be loaded with information from non-posted cycle as taught by Hagan.

The motivation for doing so would have been because Hagan teaches that **["The significant instructions generally translate into increased processing time, slowing the response of the processor determining whether the current queue entry is empty. Therefore, it would be advantageous to have an improvement with an apparatus for reducing the processing overhead for multiple processor or embedded processor architectures in posting events or tasks to a queue"(col. 1, lines 49-56)]**.

Therefore, it would have been obvious to combine Hagan and Bernasconi for the benefit of creating a patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip to obtain the invention as specified in claims 2, 5, and 18.

7. As per **claims 3, 6 and 9**, Bernasconi discloses "wherein the patch module **(patching circuitry22)** can send a modified non-posted cycle, **(which will be the new cycle),** to a destination device **(see fig. 1 and col. 9, line 57 to col. 10, line 5)**.

8. As per **claims 4, 7, 13, and 23**, Bernasconi discloses "wherein the patch module can return a completion associated with the modified non-posted cycle to the requesting

device **(16) (see fig. 1 and col. 9, line 57 to col. 10, line 5).**

9. As per **claims 11 and 21**, Bernasconi discloses a method **(system 10 of fig. 1)** comprising: "receiving an incoming request cycle **(the DSP program address, as it read,,-see col. 9, line 54)** from a requesting device **(DSP 16 of fig. 1)**; determining if the received incoming request cycle matches one or more of preprogrammed trigger conditions **(the comparator 42 of the patching circuitry 22, as discloses in col. 7, lines 16-19) (col. 9, lines 51-57, teaches that the trigger-matching logic is the logic that doest the matching between the current DSP program address and a break address correspond to a flawed DSP program stored in the ROM 18, such as at section 18b)**; determining if the received incoming cycle is a non-posted request cycle **(the trigger-matching logic automatically receives "non-posted request cycle". The "non-posted request cycle" is request that need to be complete, as stated by the applicant's specification in paragraph 0017)**; "if the received incoming request cycle matches a trigger condition and is a non-posted request cycle, loading one of the following (1) unmodified header information **(the address of the unmodified DSP program, as stated in col. 9, lines 51-57)** from the captured non-posted request cycle **(col. 8, lines 3-11 discloses "When the patch control module 60 is enabled over line 64, and the comparator 42 detects a match, the signal indicating a match over line 62 causes the patch control module 60 to initiate a patch sequence. Initiation of a patch sequence will render appropriate signals over the patch control outputs labeled patch_cycle 1 on line 74 and patch_cycle 2 on line 76. Details of patch sequence initiation will be discussed later with respect to device operation". This 'Operation', as sated in col. 9, line 57 to col. 10, line 5, discloses, "At this stage, the patching**

circuitry 22 sends to the DSP 16 a branch op code followed by a branch address corresponding to the beginning of a block of corrected DSP program software in the RAM 20 such as in section 20b thereof. Therefore, in the third general phase of device operation, the DSP 16 fetches corrected DSP program software instructions from the RAM 20 such as section 20b thereof. Moreover, corrected DSP program software can be set up to cause the DSP 16 to establish new break and branch addresses, thereby setting up additional patches to be executed during the operation of the device's DSP 16. Also, a section of corrected DSP program software typically includes, at the end thereof, jumping instructions causing the DSP 16 to recommence fetching DSP program software from the ROM 18 in a section such as 18c located downstream of the previous flawed portion 18b in the ROM 18"), (2) modified header information associated with a modified non-posted cycle, or (3) header information associated with a new request cycle generated in response the received incoming request cycle; instructing whether or not to discard a completion packet received from a designated end-device **(DSP 16 of fig. 1)** if header information is loaded with from one of the modified, non-posted request cycle and the generated, new request cycle **(directionally, fig. 1 shows the DSP program address going to the DSP 16,,,the claim language doesn't preclude the DSP 16 from being 'a requesting device' and 'an end-device'. The claim language doesn't specifically express that the requesting device is different from an end-device. Once an incoming cycle is received from a requesting device, it is considered complete until an error or a fault is found in it; as it goes through the patch circuitry 22, it will be discard because of the error or the fault. Therefore, this limitation is inherently met: see col. 9, line 51 to col. 10, line 5).**

Bernasconi fails to specifically teach specifying whether the I/O controller including a patch module **(patching circuitry22)** coupled to a completion queue is to be loaded with information from non-posted cycle. Hagan discloses a RAM or ROM wherein a completion queue is to be loaded with information from non-posted cycle" **(see abst. and col. 2, lines 50-65). See abst. and col. 4, lines 47-59, which also discloses an interrupt state to be cleared when the entry is posted, the interrupt state being associated with each entry in the queue.**

Bernasconi's invention and Hagan's invention's are analogous art because they are from the same field of endeavor storing data into queue from a processor. In view of such teaching, at the time of the invention it would have been obvious to a person of ordinary skill in the art to modify the patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip as taught by Bernasconi which includes a DSP and a patching circuitry that's consist of a trigger-matching logic, further include a control logic coupled to the trigger-matching logic is modified to include a completion queue to be loaded with information from non-posted cycle as taught by Hagan.

The motivation for doing so would have been because Hagan teaches that **["The significant instructions generally translate into increased processing time, slowing the response of the processor determining whether the current queue entry is empty. Therefore, it would be advantageous to have an improvement with an apparatus for reducing the processing overhead for multiple processor or embedded processor architectures in posting events or tasks to a queue"(col. 1, lines 49-56)]**.

Therefore, it would have been obvious to combine Hagan and Bernasconi for the benefit of creating a patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip to obtain the invention as specified in claims 11 and 21.

10. As per **claims 12, 14, and 22**, Bernasconi discloses "generating a modified non-posted, **which is a new request cycle**, in response to a matched trigger condition **(col. 9, lines 9, lines 57 to col. 10, line 5, teaches the that the control logic is taught by Bernasconi's patching circuitry 22, which carries out the steps of sending a branch op code to the DSP 16 followed by a branch address corresponding to the beginning of a block of corrected DSP program software in the RAM 20 such as in section 20b.)**; sending the modified non-posted cycle to a destination device **(processor 16 in fig. 1)**.

11. As per **claims 15**, Bernasconi discloses "discarding a completion associated with the new request cycle received from the destination device" **(see col. 10, lines 50-60)**.

12. As per **claim 16**, Bernasconi discloses a patch module **(system 10 of fig. 1)** comprising : a trigger-matching logic **(the comparator 42 of the patching circuitry 22, as discloses in col. 7, lines 16-19) (col. 9, lines 51-57, teaches that the trigger-matching logic is the logic that doest the matching between the current DSP program address and a break address correspond to a flawed DSP program stored in the ROM 18, such as at section 18b)** to capture an incoming read/write request cycle **(the DSP program address, as it read,,-see col. 9, line 54)** from an upstream device **(directionally, fig. 1 shows the DSP program address**

coming from the DSP 16) and to determine if the captured incoming read/write request cycle matches at least one trigger condition of one or more trigger condition of one or more of trigger conditions **(see col. 9, lines 51-57)**; and a control logic **(the patch control 60 of the patching circuitry 22, as discloses in col. 7, lines 44 and 45- see also col. 8, lines 4-7)** coupled **(see fig. 3, which shows the comparator 42, the trigger-matching logic, being coupled to the patch control 60)** to the trigger-matching logic to select a set of instructions upon detection of at least one matched trigger condition and to execute operations as specified by the selected set of instructions, wherein if the captured request cycle that caused a trigger is a non-posted request cycle, loading one of the following (1) unmodified header information **(the address of the unmodified DSP program, as stated in col. 9, lines 51-57)** from the captured non-posted request cycle **(col. 8, lines 3-11 discloses "When the patch control module 60 is enabled over line 64, and the comparator 42 detects a match, the signal indicating a match over line 62 causes the patch control module 60 to initiate a patch sequence. Initiation of a patch sequence will render appropriate signals over the patch control outputs labeled patch_cycle 1 on line 74 and patch_cycle 2 on line 76. Details of patch sequence initiation will be discussed later with respect to device operation". This 'Operation', as sated in col. 9, line 57 to col. 10, line 5, discloses, "At this stage, the patching circuitry 22 sends to the DSP 16 a branch op code followed by a branch address corresponding to the beginning of a block of corrected DSP program software in the RAM 20 such as in section 20b thereof. Therefore, in the third general phase of device operation, the DSP 16 fetches corrected DSP program software instructions from the RAM 20 such as section 20b thereof. Moreover, corrected DSP program software can be set up to cause the DSP 16 to establish new break and**

branch addresses, thereby setting up additional patches to be executed during the operation of the device's DSP 16. Also, a section of corrected DSP program software typically includes, at the end thereof, jumping instructions causing the DSP 16 to recommence fetching DSP program software from the ROM 18 in a section such as 18c located downstream of the previous flawed portion 18b in the ROM 18"), (2) modified header information associated with a modified non-posted cycle, or (3) header information associated with a new request cycle generated in response the received incoming request cycle; instructing whether or not to discard a completion packet received from a designated end-device **(DSP 16 of fig. 1)** if header information is loaded with from one of the modified, non-posted request cycle and the generated, new request cycle **(directionally, fig. 1 shows the DSP program address going to the DSP 16,,,the claim language doesn't preclude the DSP 16 from being 'a requesting device' and 'an end-device'. The claim language doesn't specifically express that the requesting device is different from an end-device. Once an incoming cycle is received from a requesting device, it is considered complete until an error or a fault is found in it; as it goes through the patch circuitry 22, it will be discard because of the error or the fault. Therefore, this limitation is inherently met: see col. 9, line 51 to col. 10, line 5).**

Bernasconi fails to specifically teach specifying whether the I/O controller including a patch module **(patching circuitry22)** coupled to a completion queue is to be loaded with information from non-posted cycle. Hagan discloses a RAM or ROM wherein a completion queue is to be loaded with information from non-posted cycle" **(see abst. and col. 2, lines 50-**

65). See abst. and col. 4, lines 47-59, which also discloses an interrupt state to be cleared when the entry is posted, the interrupt state being associated with each entry in the queue.

Bernasconi's invention and Hagan's invention's are analogous art because they are from the same field of endeavor storing data into queue from a processor. In view of such teaching, at the time of the invention it would have been obvious to a person of ordinary skill in the art to modify the patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip as taught by Bernasconi which includes a DSP and a patching circuitry that's consist of a trigger-matching logic, further include a control logic coupled to the trigger-matching logic is modified to include a completion queue to be loaded with information from non-posted cycle as taught by Hagan.

The motivation for doing so would have been because Hagan teaches that **["The significant instructions generally translate into increased processing time, slowing the response of the processor determining whether the current queue entry is empty. Therefore, it would be advantageous to have an improvement with an apparatus for reducing the processing overhead for multiple processor or embedded processor architectures in posting events or tasks to a queue"(col. 1, lines 49-56)]**.

Therefore, it would have been obvious to combine Hagan and Bernasconi for the benefit of creating a patch mechanism used to detect and workaround defects and conditions existing in an integrated circuit chip to obtain the invention as specified in claim 16.

13. As per **claim 17**, Bernasconi discloses "the system of claim 1,"**[see rejection to claim 1 above]** "wherein the trigger-matching logic and the control logic are incorporated within an Input/Output (I/O) chip **(with respect to this limitation,**

Bernasconi discloses the logic that doest the matching between the current DSP program address and a break address take place inside the patching circuitry 22 that is discloses in fig. 2. The patching circuitry 22 is shown in fig. 1 coupled to a control logic, incorporated within an Input/output (I/O) integrated circuit chip". (see fig. 1).

14. As per **claims 18 and 20**, Bernasconi discloses "wherein the control logic generates a modified non-posted, **which is a new request cycle**, based on the at least one matched trigger condition **(col. 9, lines 9, lines 57 to col. 10, line 5, teaches the that the control logic is taught by Bernasconi's patching circuitry 22, which carries out the steps of sending a branch op code to the DSP 16 followed by a branch address corresponding to the beginning of a block of corrected DSP program software in the RAM 20 such as in section 20b.);** sending the modified non-posted cycle to a destination device **(processor 16 in fig. 1)**.

IV. RELEVANT ART CITED BY THE EXAMINER

15. The following prior art made of record and not relied upon is cited to establish the level of skill in the applicant's art and those arts considered reasonably pertinent to applicant's disclosure. See MPEP 707.05(c).

16. The following references teach a computer system used to detect, transfer data, workaround defects and conditions existing in an integrated circuit chip.

U.S. PATENT NUMBER

US 2004/0237009
US 6,463,549
US 6,314,024

V. CLOSING COMMENTS

Conclusion

a. STATUS OF CLAIMS IN THE APPLICATION

17. The following is a summary of the treatment and status of all claims in the application as recommended by **M.P.E.P. 707.07(i):**

a (1) CLAIMS REJECTED IN THE APPLICATION

18. Per the instant office action, claims 1-18 and 20-24 have received a first action on the merits and are subject of a first action non-final.

b. DIRECTION OF FUTURE CORRESPONDENCES

19. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ernest Unelus whose telephone number is (571) 272-8596. The examiner can normally be reached on Monday to Friday 9:00 AM to 5:00 PM.

IMPORTANT NOTE

20. If attempts to reach the above noted Examiner by telephone are unsuccessful, the Examiner's supervisor, Mr. Donald Sparks, can be reached at the following telephone number: Area Code (571) 272-4201.

The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300. Information regarding the status of an application may be

obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PMR system, see her//pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217- 91 97 (toll-free).

January 11, 2006

Ernest Unelus
Patent Examiner
Art Unit 2181

DONALD SPARKS
SUPERVISORY PATENT EXAMINER